4К

УДК 519.711

В.А. Козловский, А.Ю. Максимова
Институт прикладной математики и механики НАН Украины, г. Донецк
lunaplus@mail.ru, kozlovskii@iamm.ac.donetsk.ua

Решение задачи распознавания по нечетким портретам классов

В статье предложен алгоритм формирования нечетких портретов классов образов и алгоритм нечеткого вывода для задачи распознавания образов. Формально нечеткие портреты представлены лингвистическими переменными. Предложено семантическое правило для определения функций принадлежности терм-множеств. В основу построения функций принадлежности положен частотный анализ множества прецедентов. Обобщающая способность нечетких портретов зависит от параметров построения функций принадлежности.

Введение

Анализ данных является неотъемлемой частью при решении любой практической задачи. Поэтому при решении задач распознавания образов, синтезированных по конечным выборкам прецедентов, всегда необходим предварительный анализ данных. Результат такого анализа определяет путь решения задачи.

Сформировалось достаточно много различных подходов к проблеме распознавания образов. Таковыми являются дискриминантный анализ, нейросетевой подход, построение деревьев решений, применение нечетких классификаторов и т.п. [1-3]. Каждый подход дает свои преимущества и может быть применен с модификациями для большинства практических задач. Используя различные подходы, можно получать подобные по своей структуре алгоритмы. Комплексный подход при создании классификаторов позволит соединить преимущества каждого из выбранных подходов.

Целью работы является разработка алгоритма распознавания образов, который принимает решение о принадлежности предложенного образа классу образов путем сравнения его с нечетким портретом этого класса. Такие нечеткие портреты строятся на основе предварительного анализа частоты встречаемости признаков и позволяют определить степень соответствия предложенного образа каждому из классов. Качество полученного алгоритма зависит от параметров функций, определяющих нечеткие портреты. Необходимо получить значения параметров, обеспечивающих высокую обобщающую способность таких портретов классов образов.

Алгоритм является развитием предложенного в [4] подхода.

Особенности рассматриваемой задачи и основная идея алгоритма

Значительная часть задач распознавания образов опирается на модель образа в виде вектора параметров, измеряемых или наблюдаемых. Ниже будет рассмотрена такая прикладная задача.

В этом случае классическая **постановка задачи**, которой мы будем придерживаться, выглядит следующим образом. Образы представляются векторами $x^{(i)} \in X \subset R^n$.

Задано множество классов образов V = {v_i}, I = 1,…,k и обучающая выборка Y как множество пар $Y = \{(x^{(i)}, v^{(i)}), x^{(i)} \in X^m, v^{(i)} \in V, i = 1,...,n\}$.

Предполагается, что известный набор признаков в общем случае не всегда обеспечивает полную разделимость классов, и для некоторых предъявляемых образов возникает неопределенность их отнесения к какому-то из заданных классов. Это естественным образом приводит к рассмотрению результата работы алгоритма в виде нечеткого множества $\tilde{y}(\bar{x}) = \sum_{i=1}^{k} \mu_i(\bar{x}) / v_i$, где $\mu_i(\bar{x})$ – степень принадлежности образа $\bar{x}$ классу v_i. Здесь и далее обозначения соответствуют введенным в классической теории нечетких множеств [5]. Такое представление позволит формализовать понятие высказывания: «насколько сильно похож образ на представителей данного класса образов». Получить нечеткое множество как результат задачи распознавания удобно с помощью алгоритма нечеткого вывода.

Отметим особенности задачи, которые могут быть использованы для улучшения качества алгоритма.

Исходные данные не гомогенны. Объединенные в один класс образы могут описываться параметрами, подверженными сезонным и технологическим колебаниям. В частности, в рассматривающейся ниже задаче о распознавании видов топлива и их производителей значения параметров образов испытывают колебания в зависимости от времени года, смены технологических условий и т.п. Помимо этого данные могут содержать ошибки, обусловленные человеческим фактором при их внесении в базу данных, что влечет необходимость проверки на наличие выбросов в обучающей выборке.

Основная идея алгоритма распознавания образов состоит в представлении исходной информации о классах в виде их нечетких портретов. Такие портреты формируются как совокупность лингвистических переменных. Каждая лингвистическая переменная описывает информативный признак и поименована в соответствии с именем этого признака. Терм-множества лингвистических переменных строятся в результате анализа частотных характеристик каждого из признаков по каждому классу.

В работе предложен алгоритм нечеткого вывода, с помощью которого принимается решение о принадлежности образа одному из классов. В ситуациях, когда рассматриваемый образ попадает на границы классов, ответ рассматривается в виде нечеткого множества степеней соответствия всем рассматриваемым классам с перечнем наиболее близких классов образов. Принятие решения осуществляется на основе базы правил нечетких продукций, которая строится по нечетким портретам. Настройка базы правил выполняется за счет настройки параметров функций принадлежности терм-множеств лингвистических переменных. Особенности этой настройки будут рассмотрены ниже и напрямую связаны с методом построения функций принадлежности.

Рассмотрим процесс формирования нечетких портретов более детально. В качестве информативных признаков для формирования модели алгоритма берутся признаки с низкой попарной корреляцией. В связи с этим данные по каждому показателю рассматриваются независимо друг от друга.

Сгруппируем значения элементов обучающей выборки Y по принадлежности классам образов, как показано в табл. 1, где P = $\{P_1,...,P_i,...,P_n\}$ – множество признаков, V = $\{v_1,...,v_j,...,v_k\}$ – множество классов образов. Определим множество $D_{i,j}$ как множество значений признака i для класса j: если $v_j = \{\bar{x}_{j_1}, \bar{x}_{j_2} ..., \bar{x}_{j_t}\}$, $pr_i(x_1,...,x_i,...,x_n) = x_i$ и $pr_i\, v_j = \{pr_i\, \bar{x}_{j_1}, pr_i\, \bar{x}_{j_2},..., pr_i\, \bar{x}_{j_t}\}$, то $D_{ij} = pr_i\, v_j$.

Таблица 1

Y	P_1	$\dots$	P_i	$\dots$	P_n
v_1	D_{11}	$\dots$	$D_{1,i}$	$\dots$	$D_{1,n}$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
v_j	$D_{j,1}$	$\dots$	$D_{j,i}$	$\dots$	$D_{j,n}$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
v_k	$D_{k,1}$	$\dots$	$D_{k,i}$	$\dots$	$D_{k,n}$

Каждому признаку поставим в соответствие лингвистическую переменную $L_i = \{имя(P_i), T_i, U_i, G, M\}, T_i = \{(\mu_{i,j}(x)), j = 1,..., K\}$, U_i – область определения элементов $P_i, \mu_{ij}(pr_i\,\bar{x}) \in [0,1]$ – функция принадлежности, определяющая степень уверенности, с которой образ $\bar{x}$ относится к классу образов v_j. Синтаксическое правило *G*, порождающее названия переменных, в данном случае тривиально, т.к. все термы атомарные, и заключается в присвоении функции принадлежности имени класса, который она представляет. Семантическое правило *M* представлено в виде алгоритма формирования функций принадлежности и будет подробно рассмотрено дальше. Функции принадлежности $\mu_{i,j}$ можно строить по-разному, и в зависимости от способа построения будет меняться обобщающая способность, которая определяется коэффициентами α и β, описанными ниже.

Определение. Нечетким портретом S_j называется множество значений лингвистических переменных, соответствующих классу v_j: $S_j = \{\mu_{ji}\}, i = 1,..., N$.

В табл. 2 представлена структура лингвистических переменных, знак $\mapsto$ обозначает соответствие.

Таблица 2

Y	L_1	$\dots$	L_i	$\dots$	L_n
$v_1 \mapsto S_1$	$\mu_{11}^{\alpha,\beta}$	$\dots$	$\mu_{1i}^{\alpha,\beta}$	$\dots$	$\mu_{1n}^{\alpha,\beta}$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$v_j \mapsto S_j$	$\mu_{j1}^{\alpha,\beta}$	$\dots$	$\mu_{ji}^{\alpha,\beta}$	$\dots$	$\mu_{jn}^{\alpha,\beta}$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$v_k \mapsto S_k$	$\mu_{k1}^{\alpha,\beta}$	$\dots$	$\mu_{ki}^{\alpha,\beta}$	$\dots$	$\mu_{kn}^{\alpha,\beta}$

Метод формирования функций принадлежности нечеткого портрета

Рассмотрим метод построения функций принадлежности. Фактически в этой части определяется семантическое правило для получения значений лингвистических переменных в виде алгоритма, основанного на анализе частоты встречаемости признаков.

Ранее были определены множества D_{ij}. В результате анализа этих множеств строятся функции принадлежности μ_{ij}. Далее условимся опускать индексы i, j при описании алгоритма, учитывая единообразие в построении всех функций.

Алгоритм основан на концепции скользящего окна и является расширением подхода, используемого при построении гистограмм в статистике [6]. Использование гистограмм позволяет осуществлять поиск выбросов.

Результатом работы алгоритма будет кусочно-линейная функция μ, которая соответствует полигону частот в терминах статистики в предельном случае, когда ширина скользящего окна равна шагу смещения окна.

Определим среднее расстояние h_{cp} между точками множества D по формуле $h_{cp} = \dfrac{\max(D)-\min(D)}{|D|-1}$, где $|D|$ – мощность множества. Шаг смещения окна зададим как $step = вh_{cp}$, а размер скользящего окна как $wind = гh_{cp}$. Обозначим $\alpha = \dfrac{step}{wind}$. Коэффициенты α, β, γ определяют вид функции принадлежности.

На рис. 1 представлена схема применения скользящего окна.

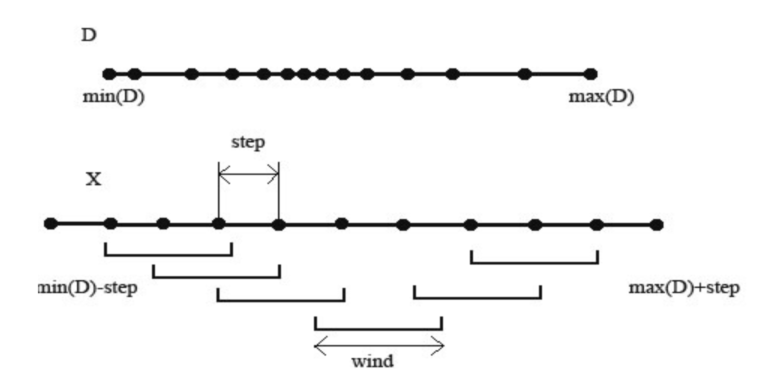


Рисунок 1 – Схема применения скользящего окна

Область определения функции принадлежности $\mu(x)$ задается как $X = [\min D - step, \max(D)+step]$. В этом множестве выделяются базовые точки, задаваемые шагом смещения скользящего окна $step$. Вычисляется значение функции принадлежности в каждой такой точке. В промежуточных точках полагаем, что значение функции определяется точкой, лежащей на прямой, соединяющей две соседние базовые точки. Окончательно нормируем сформированную функцию, деля ее на максимальное значение. В результате получаем функцию принадлежности $\mu^{\alpha,\beta}(x)$, значения которой принадлежат интервалу $[0,1]$.

Отношение $\alpha = \dfrac{\beta}{\gamma}$ определяет вид графика и обобщающую способность нечетких портретов. Отметим следующие особенности, влияющие на конечные функции принадлежности нечетких портретов и свойства алгоритма распознавания:

– при малом числе образов данного класса в обучающей выборке велико число смен знаков производной функции (обобщающая способность функций ухудшается, объем данных недостаточен);

– чем меньше коэффициент $\alpha \in (0,1]$, тем более «жесткими» являются функции принадлежности с точки зрения оценки степеней соответствия образа классу;

– с ростом коэффициента $\beta \in [1,...,5]$ функции становятся более гладкими, уменьшается число смен знаков их производных.

Настройка коэффициентов α, β проводится по серии экспериментов. В них определяются значения, при которых достигается наилучшая различающая способность алгоритма распознавания. Найденные значения α, β можно считать результатом обучения алгоритма.

Алгоритм нечеткого вывода

Как было сказано ранее, основной идеей предложенного алгоритма является сопоставление предложенного образа x с нечеткими портретами классов образов и вычисление степени соответствия образа каждому из классов. Будем использовать преимущества систем нечеткого вывода и представим процесс принятия решения в виде алгоритма нечеткого вывода.

Для этого будем следовать основной логике алгоритмов нечеткого вывода, которые включают в себя следующие этапы: формирование базы правил, фаззификации, агрегирования, активации и аккумуляции [7].

На диаграмме активностей (рис. 2) поэтапно представлен рассматриваемый алгоритм нечеткого вывода. Процедуры активации и аккумуляции, которые выполняются в классических алгоритмах нечеткого вывода, здесь отсутствуют.

Результатом процедуры агрегации является дискретное нечеткое множество $\tilde{y}$. В ситуациях, когда существует явный лидер, т.е. класс образов с достаточно большой степенью уверенности, и такой класс один, выполняется дефаззификация, и этот класс-лидер становится результатом работы алгоритма распознавания образов.

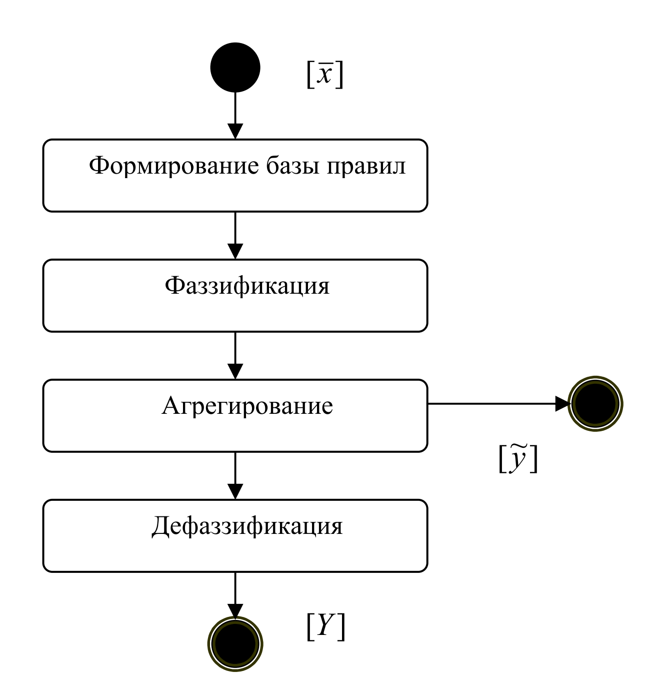


Рисунок 2 – UML-диаграмма активностей «Алгоритм нечеткого вывода».

Рассмотрим детально шаги алгоритма нечеткого вывода.

Как уже говорилось выше, база правил строится по нечетким портретам. В системах нечеткого вывода база правил нечетких продукций представляется набором правил, состоящих из нечетких лингвистических высказываний вида: «L_i *есть* v_j», где L_i – наименование лингвистической переменной, а v_i – ее значение, которому соответствует отдельный лингвистический терм из базового терм-множества лингвистической переменной L_i.

Запишем правила нечетких продукций через определенные ранее лингвистические переменные.

ПРАВИЛО «S_1» :

ЕСЛИ «L_1 *есть* v_1» И… И L_i *есть* v_1 И … И L_n *есть* v_1 ТО Y *есть* (v_1 / μ_{v_1});

…

ПРАВИЛО «S_j» :

ЕСЛИ «L_1 *есть* v_j» И… И L_i *есть* v_j И … И L_n *есть* v_j ТО Y *есть* (v_j / μ_{v_j});

…

ПРАВИЛО «S_k» :

ЕСЛИ «L_1 *есть* v_k» И… И L_i *есть* v_k И … И L_n *есть* v_k ТО Y *есть* (v_k / μ_{v_k}).

Таким образом, для каждого нечеткого портрета S_j строится правило нечеткого вывода.

Пусть на вход алгоритма приходит $\bar{x}$. Этап фаззификации заключается в вычислении значений $\mu_{ij}(\bar{x}) = \mu_{ij}(pr_i(\bar{x}))$.

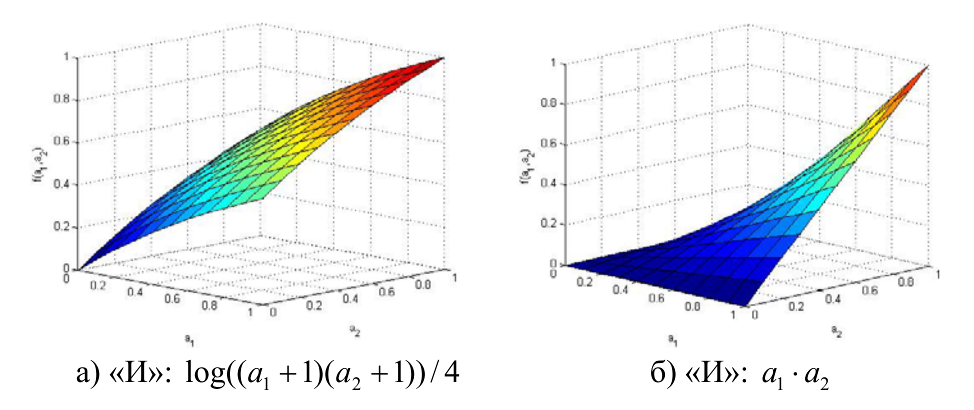
Особенность этапа агрегирования заключается в выборе операции «И».

В качестве такой операции в системах нечеткого вывода используют различные функции, например, операции минимума или умножения.

В данном случае была использована n-местная операция «И», которая задается следующей функцией:

$$f(a_1, a_2, ..., a_n) = \log((a_1+1)(a_2+1)\cdot ... \cdot(a_n+1))/n\,,\ a_i \in [0,1] \Rightarrow f(\bar{a}) \in [0,1].$$

На рис. 3 а) показана поверхность принятия решения для такой двуместной операции *f*.



а) «И»: $\log((a_1+1)(a_2+1))/4$ б) «И»: $a_1 \cdot a_2$

Рисунок 3 – Поверхность принятия решения для двуместной операции «И»

Такая операция позволяет принимать более качественные решения при малых значениях функций принадлежности. При операции «И» типа умножения, поверхность принятия решения которой представлена на рис. 3 б), в аналогичных ситуациях возникает опасность недооценки малых значений нечетких характеристик.

Дефаззификация выполняется по формуле $v = \arg\limits_{v} \max\limits_{j} \{\tilde{y}\}$.

Оценка качества алгоритма и выбор коэффициентов α,β

Для оценки качества и стабильности алгоритма использовалась процедура скользящего контроля. В [8] предложены функционалы, которые характеризуют обобщающую способность алгоритма μ по конечной совокупности объектов X^L –

функционал полного скользящего контроля Q_c и функционал среднего отклонения частоты ошибок на контроле от частоты ошибок на обучении Q_d:

$$Q_c(\mu, X^L) = \frac{1}{N}\sum_{n=1}^{N} \nu(\mu(X_n^l), X_n^k),$$

$$Q_d(\mu, X^L) = \frac{1}{N}\sum_{n=1}^{N} (\nu(\mu(X_n^l), X_n^k) - \nu(\mu(X_n^l), X_n^l)),$$

где $(X_n^l, X_n^k), n = 1,...,N$ – всевозможные разбиения выборки X^L на обучающую и контрольную, $L = l + k$; $\nu(\mu, X^L)$ – частота ошибок алгоритма μ на обучающей выборке X^L.

Результат процедуры скользящего контроля отличается в зависимости от способа формирования указанного множества разбиений. В [9] экспериментально доказано, что для выбора наилучшего классификатора *k*-кратный скользящий контроль со значением *k*, равным 10, лучше, чем более дорогостоящая оценка с одним отделяемым объектом. *K*-кратным скользящим контролем называется вариант скользящего контроля, в котором множество разбиений образуется *k*-непересекающимися контрольными выборками. Если используются все разбиения с контрольной выборкой единичной длины, то такая процедура скользящего контроля называется оценкой с одним отделяемым объектом.

По этой причине в данной работе для тестовой задачи использовался десятикратный скользящий контроль.

На практике скользящий контроль используют и для оптимизации небольшого числа параметров, определяющих структуру алгоритма. Для предложенного в работе алгоритма такими параметрами являются коэффициенты α, β. Анализ функционалов качества Q_c, Q_d позволяет определить α, β, обеспечивающие хорошую разделяющую способность.

Решение практической задачи

Предложенный алгоритм был апробирован на реальной задаче распознавания производителей топлива и их видов. Образ определяется набором из пяти измеряемых показателей, таких как октановое число, суммарные ароматические соединения, ароматические бензолы, олефины и метил-трет-бутиловый эфир (МТБЭ). Число классов равно шести и границы классов частично пересекаются.

Для каждого класса образов строился нечеткий портрет. Для различных $\alpha \in [0.05,1]$, $\beta \in [1,8]$ были получены нечеткие портреты. Для оценки качества алгоритма использовалась процедура скользящего контроля. Наилучший результат получен для $\alpha = 0,15$, $\beta = 7$. Ошибка распознавания не превышала 7%.

Выводы

В работе введено понятие нечеткого портрета и предложен алгоритм распознавания образов на основе сопоставления рассматриваемого образа с нечеткими портретами классов образов. Параметры нечеткого портрета настраиваются подбором специальных коэффициентов α, β, определяющих общий вид функций принадлежности, входящих в нечеткий портрет, и их обобщающую способность. Найденные экспериментально параметры α, β обеспечивают более высокое качество распознавания по сравнению с остальными значениями.

Принятие решения в задаче распознавания образов осуществляется алгоритмом нечеткого вывода, база правил нечетких продукций которого строится автоматически – в результате анализа обучающей выборки. Предложен метод построения функций принадлежности нечетких портретов.

Для контроля качества полученного алгоритма использована процедура скользящего контроля. Методом скользящего контроля были оценены качество и обобщающая способность полученного алгоритма. Алгоритм применен для решения практической задачи распознавания различных видов топлива.

Литература

1. Хант Э. Искусственный интеллект / Хант Э. ; пер. с англ. – М. : Мир, 1978. – 558 с.
2. Ту Дж. Принципы распознавания образов / Дж. Ту, Р. Гонсалес ; пер. с англ. – М. : Мир, 1978. – 415 с.
3. Барсегян А.А. Методы и модели анализа данных : OLAP и Data Mining : [учебное пособие] / А.А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод. – СПб. : БХВ-Петербург, 2004. – 336 с.
4. Козловский В.А. Алгоритм распознавания, основанный на нечетком подходе / В.А. Козловский, А.Ю. Максимова // Искусственный интеллект. – 2008. – № 4. – С. 594-599.
5. Заде Л. Понятие лингвистической переменной и его применение к принятию приближенных решений / Л. Заде ; под ред. Н.Н. Моисеева и С.А. Орловского. – М. : Мир, 1976. – 168 с.
6. Афифи А.Статистический анализ. Подход с использованием ЭВМ // А. Афифи, С. Эйзен. – М. : Мир, 1982. – 289 с.
7. Леоненков А.В. Нечеткое моделирование в среде MATLAB и fuzzyTECH / А.В. Леоненков. – СПб. : БХВ-Петербург, 2003. – 736 с.
8. Воронцов К.В. Комбинаторный подход к оценке качества обучаемых алгоритмов / К.В. Воронцов // Математические вопросы кибернетики. Вып. 13 : сборник статей ; под ред. О.Б. Лупанова. – М. : Физматлит, 2004. – С. 5-36.
9. Kohavi R.A. A study of cross-validation and bootstrap for accuracy estimation and model selection [Электронный ресурс] / R.A. Kohavi // IJCAI. – 1995. – Режим доступа: http: // citeseer.isu.psu.edu/ kahavi95study.html

В.А. Козловський, О.Ю. Максимова

Розв’язання задачі розпізнавання за нечіткими портретами класів

В статті розглянуто алгоритм формування нечітких портретів класів образів та алгоритм нечіткого висновку для задачі розпізнавання образів. Формально нечіткі портрети представлено лінгвістичними змінними. Запропоновано семантичне правило для визначення функцій приналежності терм-множин. Основою побудови функцій приналежності є частотний аналіз множин прецедентів. Здатність нечітких портретів до узагальнення залежить від параметрів побудови функцій приналежності.

V.A. Kozlovskii, A. Ju. Maksimova

Decision of Pattern Recognition Problem with Fuzzy Portraits of Classes

In the given work the algorithm of creating fuzzy portraits and fuzzy inference is suggested for pattern recognition problem. Formally fuzzy portraits are represented like linguistic variables. There is semantic rule for creating membership functions of terms-set. The base of membership functions creating is frequency analysis of set precedents. The ability to generalization of fuzzy portraits is depended on membership functions creating parameters.

Статья поступила в редакцию 19.07.2010.